ECE6332 – VLSI

"LOW-POWER AND HIGH-PERFORMANCE ARITHMETIC LOGIC UNITS"

FIRST DESIGN REVIEW

ELAHEH SADREDINI

LUONAN WANG

LUIS LOPEZ RUIZ

OCTOBER 7, 2014

INTRODUCTION

In current electronic systems, size, energy efficiency, and speed have become the most important factors to be considered when designing a new system whether it is analog or digital. Given this tendency and knowing the importance of having high-efficiency basic building blocks such as adders and multipliers in digital electronics, the topic selected for this work was the design of low power and high performance arithmetic units.

In the current document, the work that has been done at this point is presented. As part of this work, several papers are summarized, and some simulations of the new structures from those papers are included.

Finally, the goals for the Proposal Presentation are stated, as well as the final goals of this project.

PUBLICATIONS SUMMARIES

[1] Yajuan He, Chip-Hong Chang and Jiangmin Gu, "An Area Efficient 64-bit Square Root Carry-select Adder for Low Power Applications" ,ISCAS 2005

Carry-select adder (CSA) is a good compromise between cost and performance in carry propagation adder design. In carry-select adder (CRA), two carry ripple adder is used and it causes more delay and area consumption for this kind of adder. Carry select adder and add-one circuit design which was proposed by T. Y. Chang and Hsiao eliminates one carry ripple adder from conventional carry select adder. It uses XNOR gates to calculate the sum, when carry is 1 and also uses buffer gates after carry-out signal due to capacitive loading. In this paper, the authors propose a square root carry-select adder with a new add-one circuit (new first zero detection). They uses just a NAND as a buffer functionality to improve the deriving capability. The number of transistors has decreased in the proposed CSA. Consequently, it decreases power-delay production and area in comparison to the last proposed CSA. As paper reports, the new CSA requires 44% fewer transistors than the conventional one.

[2] Basant Kumar Mohanty- Sujit Kumar Patel , "Area-Delay-Power Efficient Carry-Select adder", IEEE Transactions on Circuits and Systems, June 2014

In this paper, the logic operation involved in conventional carry select adder (CSLA) and binary to excess-1 converter(BEC)-based CSLA which was proposed by B. RamKamur and H. M. Kittur are analyzed to study the data dependence and to identify redundant logic operations. They have eliminated all the redundant logic operation presents in the conventional CSLA and proposed a new logic formulation for CSLA. At first, they have formulated the binary operation of 2 Ripple Carry Adder (RCA) and found that there are 2 equal terms that have been calculated redundantly. These redundant logic operation can be removed to have an optimized design for the second RCA. They also have shown that BEC method increases data dependences in the CLSA. By removing all of dependencies between RCA's and reformulating add operation, they have designed a carry select adder with 4 serial parts. In this new design, they have eliminated all of dependencies between Ripple carry adders. They have shown that their proposed method involves nearly 35% less area-delay-product than the BEC-based SQRT-CSLA. The application-specified integrated circuit(ASIC) synthesis result shows that in average, the BEC-based SQRT-CSLA design involves 48% more ADP and consumes 50% more energy than their proposed method, for different bit-widths.

[3] T. Esther, M. Asha Rani, Dr. Rameshwar rao , "Area Optimized Low Power Arithmetic and Logic Unit", ICECT 2011

In this paper, a low-power one bit full adder is proposed by the authors. They designed the full adder with 10 transistor and shows that the power consumption had reduced. Because less transistor design have less number of power supply to ground connections. They used this adder in the implementation of ALU. T. Easter Rani and and his colleagues contend that their new design has reduced power and area more than 70% in compare to conventional design and 30% in compare to transmission gates. Finally, they showed that power consumption of 16-bit ALU with 10 transistor full adder is 1197.5 uw.

[4] M. Aguirre-Hernandez, M. Linares-Aranda, "CMOS Full-Adders for Energy-Efficient Arithmetic Applications." IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 19, No. 4, 718-72,2001

For digital electronics, arithmetic operations play an important role, and a clear example of that is the inclusion of arithmetic logic units (ALUs) in robust digital systems. Full-adders are one of the basic elements of such units and following the tendency of electronics in general, designers work in different optimizations to make them smaller, faster, and with less power consumption. The work done by Aguirre-Hernandez and Linares-Aranda (2011) presents a new internal logic structure for full-adders combined with pass-transistor logic to achieve a better performance in terms of speed and energy efficiency. The proposed new logic structure compared to the previous standard structure for full-adders shows advantages such as smaller overall propagation delays by driving the multiplexers with the carry input signal, a reduction in the capacitive load for the carry input signal, and the capability to tune up individually the sum output and carry output for specific applications. Subsequently, two novel full-adders based on this structure are introduced and they are compared to five other energy-efficient full-adders. The tests and simulations show an average PDP advantage of 80% of the new two adders over the other five with only 40% of the area. Finally, the layout for the new two full-adders using a TSMC 0.18 μm CMOS technology is shown.

[5] B. Ramkumar, H. M. Kittur, "Low-Power and Area-Efficient Carry Select Adder." IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 20, No. 2, 371-375,2012

Carry-Select Adders (CSLA) are widely used circuits to perform fast arithmetic operations for processing data. The structure of this adders is relatively simple and consists of two Ripple Carry Adders (RCA) and a multiplexer in order to select the correct sum and the correct carry out. In "Low-Power and Area-Efficient Carry Select Adder", Ramkumar and Kittur present an alternative structure for CSLA with a minimal modification at the gate-level to achieve a significantly reduction in area and power

compared to the traditional CSLA, with a slightly increase in the delay. The modification proposed by the authors is to replace the section of RPA correspondent to carry equals to 1 with a Binary to Excess-1 Converter (BCE) since this circuits require less logic gates for its implementation. A 16-bit SQRT CSLA is selected to compare both designs. First, a methodology to analyze the structure of the CSLA in terms of delay and area is described. Then, using the same methodology, the alternative CSLA is analyzed and the results show an improvement in area and power of 15% and 10.56%, respectively. Finally, the results for 8-bit, 32-bit and 64-bit CSLA are shown, allowing to notice that the trade-off between power and area optimization with delay is reduced with bit size.

[6] M.Vratonjic, B. R. Zeydel, V. G. Oklobdzija, "Low- and Ultra-Low Power Arithmetic Units: Design and Comparison." Proceedings of the 2005 International Conference on Computer Design (ICCD'05).

Previous work on analyzing integrated circuits designs to determine the best approach to achieve low power consumption has been made. This work has been often made by comparing area and total gate count, which certainly does not consider the impact of transistor sizing and supply voltage scaling on delay and energy. Vratonjic, Zeydel and Oklobdzija present a short description of the previous analysis and introduce a new methodology which takes into account energy and delay to provide a better comparison of the designs. By scaling the supply voltage, the authors create the energy-delay characteristics of 32-bit adders and define the regions of operation for low and ultra-low power. With the new methodology defined, six energy-efficient adders are analyzed and compared in four areas: delay, average energy, energy-delay product and gate count. The results of their work are organized in tables and plots for an easy understanding. The work is concluded demonstrating that a comparison made as the previous works is insufficient to determine the optimum design. Additionally, with the presented methodology, it is proven that the use of supply-voltage scaling and high-performance structures improves and reduces the energy consumption for low power and ultra-low power designs.

[7] Ahmed M. Shams, Magdy A.Bayoumi, "A Novel Low-Power Building Block CMOS Cell For Adders", 1998 IEEE

In this paper the author introduced a systematic approach to design four new high-performance and low-power full adder cores with driving capability and full voltage-swing nodes for embedded structure. The new cores are capable of full-swing operation at low supply voltages. The author first summarized various full adders with full voltage swing proposed in past literature. One category is full adders without output driving capability and the other is full adders with output driving ability.

The author proposed their novel adders based on the full adder functions. Four full adder cores were presented and ten novel adder modules were built utilizing the combination of some of the four full adder cores. Then twelve bits ripple carry adders using previously designed adder modules as basic blocks were built to evaluate the performance of the proposed ten adder modules. Module M10 was proven to be superior to the other three previous designs in transistor count with only 17 transistors per bit. It consumes 51.99% to 62.99% less power than three previous designs.

[8] Amr M.Fahim, Mohamed I.Elmasry,"Low-Power High-Performance Arithmetic Circuits and Architectures",IEEE Journal of Solid-State Circuits, vol.37, NO.1 January 2002

In this paper, the author proposed a new class of differential logic family, swing limited logic(SLL). The author mentioned that low power design is one of the most pervasive trend in digital logic design. From the equation $P_{dyn} = a_{0\to1} \cdot C_L \cdot V_{dd} \cdot V_{swing} \cdot f$, we can see dynamic power consumption is proportional to the voltage swing. So if we can manage to reduce the voltage swing, we can reduce the dynamic power.

The author then presented two implementations of SLL, short-circuit current logic (SC2L) and clock-pulse controlled logic (CPCL). The author proposed a scheme using gate pipelining in which clock signal is only provided to the first pipeline stage and then send completion signal to next stage which is generated for free. Using FA as a benchmark to compare SC2L to other logic styles shows that SC2L exhibits an order of magnitude less

energy-delay product than other logic families. However, the disadvantage of this scheme is that it places a large load on node B which is the node for precharging the NMOS logic. To solve this problem, SC2L has low driving capability, the author proposed another differential logic family, clock-pulse control logic (CPCL).

The differential logic presented in this paper can benefit our design because we are also doing lower power circuit design. Using these topologies, we may improve the energy reduction in our circuits.

[9] Krishnamurthy, R.K. , Lys, I. , Carley, L.R, "Static power driven voltage scaling and delay driven buffer sizing in Mixed Swing QuadRail for sub-1 V I/O swings", Low Power Electronics and Design, 1996, International Symposium

In this paper, the author introduced a new methodology for performing low voltage logic in a high threshold voltage CMOS fabrication process called Mixed Swing QuadRail, which has four power supplies. Previous techniques have focused on using standard CMOS logic and lowering the power supply voltage because of its quadratic influence on dynamic power consumption. The problem with this technique is that the delay will increase steeply when supply voltage are below the sum of threshold voltage of NMOS and PMOS device. The Mixed Swing QuadRail methodology addresses maximum possible voltage scaling with little or no reduction in operating speed.

The author presented a two stage Mixed Swing QuadRail circuit as an example. It consists of a logic stage and driver/buffer stage. Then they provided the power and delay models for the QuadRail circuits. From that we can get that the QuadRail circuit power dissipation and delay are both posynomial function of buffer transistor size. The author then presented a power and delay comparisons between QuadRail and static CMOS for AO222 gate. With increasing loads both QuadRail and CMOS delays increase with the same steepness, but QuadRail's power increases less steeply than CMOS due to the reduced load voltage swing.

CURRENT WORK AND FUTURE TASKS

First, we started to simulate the method which is proposed in [1]. In the design, they uses RCAs which are built with CMOS mirror topology since it is the most interesting implementation in term of its trade-off between power and delay performance. In the first step, we simulated a RCA with mirror topology and the output results are presented. They uses NAND gates as a buffer functionality. One of the challenges here is that we do not know what the cell ratio for this NAND gate is.

Generally in the papers which they tried to implement an adder, they did not consider the effect of switching activities for the internal nodes. For example in the CSAs which proposed in [1], consider t1 is the delay from Cin to the last multiplexer output which is Cout (Cin for the next stage). Consider t2 is the delay from inputs to generate sum for the first level carry ripple adder. Obviously, t1>t2 and between t1 and t2, the Cout signal can change several time which makes more switching power activities. As they use square root carry select adder, this problem get worse in the final stages (which their block size is larger).

Second, we built the schematic for a new low-power, high-speed CMOS 1 bit full adder in Cadence. It's from the schematic proposed in [7]. This cell has no short circuit power because there is no inverter in the cell. Then we did the simulation for the cell to test whether it can perform correctly. The Sum output was correct but there were some errors in the Cout. Maybe we didn't simulate that quite correctly and that's what we are going to do later on, manage to get the cell fully functional. Before the design proposal, we want to get the power consumption and delay for this cell from simulation and to compare with other cells we have simulated and find the appropriate adder cell for our later work.

Finally, from [4], we drew the schematic for one of the two 1-bit new adder designs presented in the paper. The design is formed by the structure introduced by the authors in combination with DPL logic style. The design uses OR, AND, XOR and XNOR building blocks to implement the full adder. Our first approach for this design was to

present all the transistors forming those blocks together and then simulate. Given the complexity that having crossing wires in the schematic represents, before simulating and in order to have a better understanding of the internal process, we want to build and test every block separate to put them all together at the end and now, simulate the adder as it is. This will be done before the proposal in addition to a few other tasks described next.

Before de proposal we want to complete the implementation which is proposed in [1] for 64bit and after verifying our design. Then, form the simulation, we want to figure out if there is switching activities in the internal nodes. If yes, we need to find out which signals they are, and try to find a way to avoid it or at least, decrease it.

Additionally, we want to implement the SLL (Swing Limited Logic) to our adders. This include short-circuit current logic (SC2L) and clock-pulse controlled logic (CPCL). In the paper we've read about SLL, the author implemented an 8-bit carry ripple adders and had an order of magnitude less energy-delay product than other logic families. We want to put this topology in our circuits and see whether it can reduce such amount of energy-delay product. Another thing we want to do after the design proposal is try to be familiar with cadence. Because we are going to do most of the design and simulation in cadence, it's better if we can use cadence proficiently to improve our proficiency.

To complete the work before the proposal, we want to simulate the different new adder cells and compare their performance. Once we identify the best option, we want to build an 8-bit, 16-bit or 32-bit adder and see its behavior and measure the delay and power for this device.

Our final goal, and knowing what design presents a better performance, we want to figure out if we can implement a new adder by taking advantage of the positive points and strengths of the different designs analyzed in the papers and consider the possibility to use this new structure in another basic arithmetic element, such as multipliers.